

Appendix B:

MIDI Protocol Specification

MIDI Message Formats

MIDI System Reset Message

Message Type	Function Byte 1
SYSTEM RESET	0xff (255)

MIDI Controller Channel Messages

Because MIDI restricts the Controller Channels to a range of 0-127, and the LP-X has channels with values greater than 127, the LP-X extends the channel range by defining a Controller Channel message that sets the high order channel. This High Controller Channel message need only be sent after a SYSTEM RESET message or when the high order channel changes. The LP-X saves the high order channel and uses it with subsequent Controller Channel messages to create the referenced Extended Controller Channel. A Controller Channel value of 127 in Byte 2 indicates a High Controller Channel message and the byte 3 value of the message is the new High Order Channel.

The Controller Channel message sets the value of the referenced Extended Controller Channel.

Message Type	Function Byte 1	Controller Channel Byte 2	Value Byte 3
Controller Channel	0xB0 (176) + MIDI Channel (0-15)	0-126 = Low order controller channel	0-127 = value / 2
High Controller Channel	0xB0 (176) + MIDI Channel (0-15)	127 = Set High Order Channel	0-127 = High order channel

Note: After the LP-X receives a System Reset message it clears its High Order Channel value and will send an High Controller Channel message before sending the first Controller Channel message. The sending MIDI Controller should send a High Controller Channel message before sending the first Controller Channel message.

The Controller Channel (byte 2) is the low order channel value. The actual extended controller channel referenced by the message is:

$$\text{Extended Channel} = ((\text{Last High Order Channel}) * 127) + \text{Low Order Channel}$$

The transmitted value in byte 3 is half the actual value of the channel. So a value of 2 is sent as 1 and a value of 255 is sent as 127.

MIDI Message Output

The LP-X conforms to the MIDI message format specification. The LP-X is capable of sending MIDI message which represent changes in front panel inputs used by the Realtime cue processing code. It sends messages only when the system is in the RUN mode. The LP-X does not send any MIDI messages for changes to user interface front panel inputs.

Section 1 details the message formats. The LP-X only sends MIDI Controller Channel Messages with the unit's assigned MIDI output channel number.

The LP-X sends one or two MIDI messages whenever the system detects a button press, button release or fader movement related to the running of a show. It does not send messages related to programming a show. The first message, a High Controller Channel message, is optional and defines the high order controller channel. The first message is sent only if the High Order Controller Channel changed from the last MIDI output message. Its controller function channel is set to MIDI_EXTENDED_CHANNEL (127). The second, a Controller Channel message, contains the low order controller channel and the current value for the input.

MIDI Message Input

The LP-X accepts all valid MIDI messages but only processes MIDI Controller Channel, High Controller Channel and System Reset messages. If the MIDI input channel number is set to OMNI, then the system processes all MIDI Controller Channel and High Controller Channel messages, ignoring the MIDI channel number. Otherwise the system processes only MIDI Controller Channel messages containing the unit's MIDI input channel number.

The LP-X processes MIDI High Controller Channel messages where the controller function channel is set to MIDI_EXTENDED_CHANNEL (127) by saving the high order controller channel value. It uses this value when processing MIDI Controller Channel messages that have a function channel value of 0-126 to form an Extended Controller Channel.

The LP-X always accepts the MIDI System Reset message. Upon receiving the System Reset message, the LP-X performs the following actions:

1. puts unit into RUN mode
2. shows the conventional scene channel display
3. clears all button presses
4. zeroes the internal value of all faders
5. enables bank 0 faders and buttons

Example of a MIDI Message Sequence

Assume LP-X input channel = 3

Operation	Message 1	Message 2
System Reset	0xFF	n/a
Set FADER_01 = 255 (100%)	0xB3 0x7F 0x00	0xB3 0x0D 0x7F
Set FADER_02 = 128 (50%)	0xB3 0x0E 0x40	n/a
Set FADER_BUMP_01 = ON	0xB3 0x7F 0x01	0xB3 0x03 0x7F
Set FADER_BUMP_01 = OFF	0xB3 0x03 0x00	n/a

LP-X Controller Channel Identities

Symbolic Name	ID
MASTER_FADER	2
MANUAL_FADER	3
STACK_FADER	4
PLAYBACK_00	5
PLAYBACK_01	6
PLAYBACK_02	7
PLAYBACK_03	8
PLAYBACK_04	9
PLAYBACK_05	10
PLAYBACK_06	11
FADER_01	13
FADER_02	14
FADER_03	15
FADER_04	16
FADER_05	17
FADER_06	18
FADER_07	19
FADER_08	20
FADER_09	21
FADER_10	22
FADER_11	23
FADER_12	24

Symbolic Name	ID
FADER_13	25
FADER_14	26
FADER_15	27
FADER_16	28
FADER_17	29
FADER_18	30
FADER_19	31
FADER_20	32
FADER_21	33
FADER_22	34
FADER_23	35
FADER_24	36
FADER_BUMP_VIRTUAL	154
KEYPAD_EXCEPT	183
PLAYBACK_SELECT_01	199
PLAYBACK_SELECT_02	200
PLAYBACK_SELECT_03	201
PLAYBACK_SELECT_04	202
PLAYBACK_SELECT_05	203
PLAYBACK_SELECT_06	204
PLAYBACK_BUMP_01	156
PLAYBACK_BUMP_02	157

Symbolic Name	ID
PLAYBACK_BUMP_03	158
PLAYBACK_BUMP_04	159
PLAYBACK_BUMP_05	160
PLAYBACK_BUMP_06	161
BLACK_OUT_BUTTON	111
STACK_ON_BUTTON	124
BACK_BUTTON	121
HOLD_BUTTON	122
GO_BUTTON	123
STACK_SELECT	206
FADER_BUMP_01	130
FADER_BUMP_02	131
FADER_BUMP_03	132
FADER_BUMP_04	133
FADER_BUMP_05	134
FADER_BUMP_06	135
FADER_BUMP_07	136
FADER_BUMP_08	137

Symbolic Name	ID
FADER_BUMP_09	138
FADER_BUMP_10	139
FADER_BUMP_11	140
FADER_BUMP_12	141
FADER_BUMP_13	142
FADER_BUMP_14	143
FADER_BUMP_15	144
FADER_BUMP_16	145
FADER_BUMP_17	146
FADER_BUMP_18	147
FADER_BUMP_19	148
FADER_BUMP_20	149
FADER_BUMP_21	150
FADER_BUMP_22	151
FADER_BUMP_23	152
FADER_BUMP_24	153
ENCODER_PAGE	273